

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Kirkland et al.** §
§ Confirmation No.: **5894**
Serial No. **10/660,013** §
§ Group Art Unit: **2162**
Filed: **September 11, 2003** §
§ Examiner: **Ly, Anh**
For: **Method and Apparatus for** §
Searching Universal Resource §
Identifiers §

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

35525
PATENT TRADEMARK OFFICE
CUSTOMER NUMBER

APPEAL BRIEF (37 C.F.R. 41.37)

This brief is in furtherance of the Notice of Appeal, filed in this case on October 26, 2007.

A fee of \$510.00 is required for filing an Appeal Brief. Please charge this fee to IBM Corporation Deposit Account No. 09-0447. No additional fees are believed to be necessary. If, however, any additional fees are required, I authorize the Commissioner to charge these fees, which may be required to IBM Corporation Deposit Account No. 09-0447. No extension of time is believed to be necessary. If, however, an extension of time is required, the extension is requested, and I authorize the Commissioner to charge any fees for this extension to IBM Corporation Deposit Account No. 09-0447.

REAL PARTY IN INTEREST

The real party in interest in this appeal is the following party: International Business Machines Corporation of Armonk, New York.

RELATED APPEALS AND INTERFERENCES

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

STATUS OF CLAIMS

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 1-15

B. STATUS OF ALL THE CLAIMS IN APPLICATION

1. Claims canceled: None
2. Claims withdrawn from consideration but not canceled: None
3. Claims pending: 1-15
4. Claims allowed: None
5. Claims rejected: 1-15
6. Claims objected to: None

C. CLAIMS ON APPEAL

The claims on appeal are: 1-15

STATUS OF AMENDMENTS

A Response after the Final Office Action was not filed.

SUMMARY OF CLAIMED SUBJECT MATTER

A. CLAIM 1 - INDEPENDENT

Claim 1 is directed to a method in a data processing system for searching for Web pages within a Web site. A search statement is received as a result of a user's input in which the search statement includes a universal resource identifier and a regular expression. (pg. 13, lines 13-16, pg. 14, lines 28-30, pg 15, lines 21-25, pg. 16, lines 19-25, pg. 17, lines 10-14; **Figure 4A and 4B, 404, 416, Figure 5, 500, Figure 7, 700, 702, Figure 8, 800, 802.**) A set of universal resource identifiers associated with the universal resource identifier in the request are retrieved to form a set of retrieved universal resource identifiers. (pg. 13, line 19 - pg14, line 5, pg. 16, lines 25-28; **Figure 4A 414, Figure 7, 704.**) These retrieved universal resource identifiers are parsed for the regular expression to form search results. (pg. 14, line 3-20, pg. 15, lines 13-14, pg. 16, line 29 – pg. 17, line 1, pg. 17, lines 14-17; **Figure 7, 704, 706, Figure 8, 804.**) The search results are returned in which the search results include a list of universal resource identifiers associated with Web pages for the Web site. (pg. 14, line 21-27, pg. 15, lines 15-18, pg. 17, lines 1-3, pg. 17, lines 18-23; **Figure 7, 708, Figure 8, 806, 808.**)

B. CLAIM 7 - INDEPENDENT

Claim 7 is directed to a data processing system for searching for Web pages within a Web site. The data processing system comprises a bus system, a communications unit connected to the bus system (**Figure 2, 206; Figure 3, 306**), a memory connected to the bus system (**Figure 2, 209; Figure 3, 304**), wherein the memory includes a set of instructions, and a processing unit connected to the bus system (**Figure 2, 202, 204; Figure 3, 302**). The processing unit executes the set of instructions to receive a search statement as a result of a user input in which the search statement includes a universal resource identifier and a regular expression. (pg. 13, lines 13-16, pg. 14, lines 28-30, pg 15, lines 21-25, pg. 16, lines 19-25, pg. 17, lines 10-14; **Figure 4A and 4B, 404, 416, Figure 5, 500, Figure 7, 700, 702, Figure 8, 800, 802.**) The processing unit executes the set of instructions to retrieve a set of universal resource identifiers associated with the universal resource identifier in the request to form a set of retrieved universal resource identifiers. (pg. 13, line 19 - pg14, line 5, pg. 16, lines 25-28; **Figure 4A 414, Figure 7, 704.**)

The processing unit executes the set of instructions to parse these retrieved universal resource identifiers for the regular expression to form search results. (pg. 14, line 3-20, pg. 15, lines 13-14, pg. 16, line 29 – pg. 17, line 1, pg. 17, lines 14-17; **Figure 7, 704, 706, Figure 8, 804.**) The processing unit executes the set of instructions to return the search results in which the search results include a list of universal resource identifiers associated with Web pages for the Web site. (pg. 14, line 21-27, pg. 15, lines 15-18, pg. 17, lines 1-3, pg. 17, lines 18-23; **Figure 7, 708, Figure 8, 806, 808.**)

C. CLAIM 8 - INDEPENDENT

Claim 8 is directed to a data processing system for searching for Web pages within a Web site. The data processing system comprises a receiving means (**Figure 3, 320; Figure 4A, 400 and 402**) for receiving a search statement as a result of a user input in which the search statement includes a universal resource identifier and a regular expression. (pg. 13, lines 13-16, pg. 14, lines 28-30, pg 15, lines 21-25, pg. 16, lines 19-25, pg. 17, lines 10-14; **Figure 4A and 4B, 404, 416, Figure 5, 500, Figure 7, 700, 702, Figure 8, 800, 802.**) Also, the data processing system comprises a retrieving means (**Figure 2, 218; Figure 3, 320**) for retrieving a set of universal resource identifiers associated with the universal resource identifier in the request to form a set of retrieved universal resource identifiers. (pg. 13, line 19 - pg14, line 5, pg. 16, lines 25-28; **Figure 4A 414, Figure 7, 704.**) Additionally, the data processing system comprises a parsing means (**Figure 2, 202, 204; Figure 3, 302.**) for parsing these retrieved universal resource identifiers for the regular expression to form search results. (pg. 14, line 3-20, pg. 15, lines 13-14, pg. 16, line 29 – pg. 17, line 1, pg. 17, lines 14-17; **Figure 7, 704, 706, Figure 8, 804**) Further, the data processing system comprises a returning means (**Figure 3, 320; Figure 4A, 400 and 402**) for returning the search results in which the search results include a list of universal resource identifiers associated with Web pages for the Web site. (pg. 14, line 21-27, pg. 15, lines 15-18, pg. 17, lines 1-3, pg. 17, lines 18-23; **Figure 7, 708, Figure 8, 806, 808.**)

D. CLAIM 5 - DEPENDENT

Claim 5 is directed to the method of claim 1, wherein the parsing step includes searching a table of contents for a match to the regular expression, wherein the table of contents contains

the retrieved universal resource identifiers. (pg. 13, line 26 – pg. 14, line 20, pg. 15, lines 13-16, pg. 16, lines 4-12, pg. 16, line 25 – pg. 17, line 1, pg. 17, lines 14-17; **Figure 4A, 4B 412, 414, Figure 6, Figure 7, 704, 706, Figure 8, 804**)

E. CLAIM 12 - DEPENDENT

Claim 12 is directed to data processing of claim 8, wherein the parsing means includes a searching means (**Figure 3, 320; Figure 4A, 400 and 402**) for searching a table of contents for a match to the regular expression, wherein the table of contents contains the retrieved universal resource identifiers. (pg. 13, line 26 – pg. 14, line 20, pg. 15, lines 13-16, pg. 16, lines 4-12, pg. 16, line 25 – pg. 17, line 1, pg. 17, lines 14-17; **Figure 4A, 4B 412, 414, Figure 6, Figure 7, 704, 706, Figure 8, 804**)

F. CLAIM 15 - INDEPENDENT

Claim 15 is directed to a method in a data processing system for searching for Web pages within a Web site. A search statement is received from a user at a client browser, wherein the search statement includes a universal resource identifier and a regular expression. (pg. 13, lines 13-16, pg. 14, lines 28-30, pg 15, lines 21-25, pg. 16, lines 19-25, pg. 17, lines 10-14; **Figure 4A and 4B, 404, 416, Figure 5, 500, Figure 7, 700, 702, Figure 8, 800, 802.**) In response to receiving the search statement at the client browser, a request is sent, by the client browser, to a server to retrieve a table of contents, wherein the table of contents comprises universal resource identifiers associated with the universal resource identifier in the request. (pg. 13, line 26 – pg. 14, line 20, pg. 15, lines 13-16, pg. 16, lines 4-12, pg. 16, line 25 – pg. 17, line 1, pg. 17, lines 14-17; **Figure 4A, 4B 412, 414, Figure 6, Figure 7, 704, 706, Figure 8, 804.**) The table of contents is received from the server. (pg. 14, line 3-20, pg. 16, lines 25-28; **Figure 4A, 414, Figure 7, 704.**) The universal resource identifiers in the received table of contents are parsed for the regular expression, by the client browser, to form search results. (pg. 14, line 3-20, pg. 15, lines 13-14, pg. 16, line 29 – pg. 17, line 1; **Figure 7, 704, 706.**) The search results are displayed to a user, wherein the search results include a list of universal resource identifiers associated with Web pages for the Web site. (pg. 14, line 21-27, pg. 15, lines 15-18, pg. 17, lines 1-3, pg. 17, lines 18-23; **Figure 7, 708, Figure 8, 806, 808.**)

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The grounds of rejection to review on appeal are as follows:

A. 35 U.S.C. § 102, Anticipation (Claims 1-15)

Claims 1-15 are rejected under 35 U.S.C. § 102 as being anticipated by *Schneider, Fictitious Domain Name Method, Product, and Apparatus*, U.S. Patent No. 7,136,932, November 14, 2006

Note to Board: Claim 14 stands rejected under 35 U.S.C. 101 because the bodies of claim 14 in view of MPEP 2106 (IV)(C)(2)((1) & (2) & (a) & (b) & (c)) sections are non statutory because they are lacking of real world useful result.

In view of recent case law, Appellants do not appeal this rejection; but are willing to amend this claim as appropriate to overcome the rejection.

ARGUMENT

A. 35 U.S.C. § 102, Anticipation (Claims 1-15)

The Final Office Action rejects claims 1-15 under 35 U.S.C. § 102 as being anticipated by *Schneider, Fictitious Domain Name Method, Product, and Apparatus*, U.S. Patent No. 7,136,932, November 14, 2006 (hereinafter “*Schneider*”). This rejection is respectfully traversed.

A.1. 35 U.S.C. § 102, Anticipation (Claims 1-14)

The Final Office Action states:

With respect to claim 1, Schneider teaches a method in a data processing system for searching for Web pages within a Web site (a system for searching web pages from one of search engines to locate web pages or hits within a Web site from clients (item 1 10): see fig. 1a and 1b, and col. 17, lines 34-44; also col. 10, lines 58-67), the method comprising:

receiving a search statement as a result of a user input, wherein the search statement includes a universal resource identifier and a regular expression (receiving the input search request or search or query string including URI or string of characters for identifying an abstract or physical resource from the client of the system: see fig. 2a, col. 4, lines 30-56 and col. 18, lines 30-56);

retrieving universal resource identifiers associated with the universal resource identifier in the request to form retrieved universal resource identifiers (retrieving from a database to generate valid URIs based on the search string: fig. 16 and col. 34, lines 18-32);

parsing the retrieved universal resource identifiers for the regular expression to form search results (parsing retrieved URIs via a parsing schema: see fig. 2b, item 260 and 2a, item 210: col. 21, lines 48-63; also col. 30, lines 30-42 and col. 18, lines 30-55); and

returning the search results, wherein the search results include a list of universal resource identifiers associated with the Web pages within the Web site (the result of the search is displayed (item 222 in fig. 2a) and as a list of valid URIs (fig. 13): col. 30, lines 22-30 and col. 18, lines 40-55).

Final Office Action dated August 10, 2007, pp.5-6.

With respect to this rejection, a prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d

1566, 1567 (Fed. Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218, U.S.P.Q. 781 (Fed. Cir. 1983). In this particular case, each and every feature of the presently claimed invention is not identically shown or described in *Schneider*, arranged as they are in the claims.

For example, claim 1 recites the following:

1. A method in a data processing system for searching for Web pages within a Web site, the method comprising:
 - receiving a search statement as a result of a user input, wherein the search statement includes a universal resource identifier and a regular expression;
 - retrieving universal resource identifiers associated with the universal resource identifier in the request to form retrieved universal resource identifiers;
 - parsing the retrieved universal resource identifiers for the regular expression to form search results; and
 - returning the search results, wherein the search results include a list of universal resource identifiers associated with the Web pages within the Web site.

The Final Office Action alleges that *Schneider* teaches all the features of claim 1. Appellants respectfully disagree. For example, *Schneider* fails to teach the feature of *receiving a search statement as a result of a user input, wherein the search statement includes a universal resource identifier and a regular expression*. The Final Office Action points to column 4, lines 30 through 56 and column 18, lines 29 through 56, reproduced below for the Board's convenience, as teaching this feature.

A Uniform Resource Identifier (URI) is a compact string of characters for identifying an abstract or physical resource. URIs are the generic set of all names and addresses that refer to objects on the Internet. URIs that refer to objects accessed with existing protocols are known as Uniform Resource Locators (URLs). A URL is the address of a file accessible on the Internet. The URL contains the name of the protocol required to access the resource, a domain name or IP address that identifies a specific computer on the Internet, and a hierarchical description of a file location on the computer. In addition, the last (optional) part of the URL

may be a "query string" preceded by "?" or a "fragment identifier" preceded by "#". The fragment identifier indicates a particular position within the specified file. For example the URL "http://www.example.com:80/index.html#appendix", where "http" is the scheme or protocol, "www.example.com" is the host server name or Fully Qualified Domain Name (FQDN), "80" is the port connection for the HTTP server request, "index.html" is the filename located on the server, and "appendix" is the identifier to display a specific portion of the HTML file called "index". The URL "http://www.example.com" also retrieves an HTML file called "index" on the HTTP server called "example.com". By default, when either a port or filename is omitted upon accessing a HTTP server via a URL, the client browser interprets the request by connecting via port **80**, and retrieving the HTML file called "index".

FIG. 2a is a top-level flowchart illustrating the steps of an exemplary prior art system for requesting a network resource from an identifier by using resource location and/or name resolution services. A network access apparatus **110**, servlet, applet, stand-alone executable program, command line of a device such as a phone browser, or user interface element such as a text box object or location field **154** of a web browser **112**, can receive and parse input such as text in step **210**. The input **210** can then be updated in step **212** into a history database. Tests can also be performed to determine how to process the received input **210**. For instance, when it is determined in step **214** that the input **210** has no "." delimiters or " " delimiters only, it becomes clear that there is no domain name or IP address present and the input **210** can be processed as a search request in step **218**. Results if any, can then be notified, accessed, and/or displayed in step **222**. When the presence of the "." delimiter is determined in step **214**, the input **210** may include either an IP address or a domain name. When a domain name is parsed in step **210**, the validity of the domain name is determined in step **226**. Validity of URI syntax is provided in T. Berners-Lee, "Informational RFC (Request for Comment) 1630: Universal Resource Identifiers in WWW--A Unifying Syntax for the Expression of Names and Addresses of Objects on the Network as used in the World-Wide Web", Internet Engineering Task Force (IETF), June 1994, "<http://www.faqs.org/rfcs/rfc1630.html>", which is herein incorporated by reference.

As can be seen, the passage in column 4 of *Schneider* merely teaches the definition of a universal resource identifier. The passage does not teach the limitation of *receiving a search statement as a result of a user input, wherein the search statement includes a universal resource identifier and a regular expression*. Furthermore, the passage in column 18 of *Schneider* merely teaches the steps of requesting a network resource by using resource location

and/or name resolution services. The passage details how an input is parsed to determine if it contains delimiters. If the input does contain delimiters, then a determination is made as to whether a valid IP address or domain name is present. If no delimiter is present then the input is processed as a normal search request. Thus, *Schneider* teaches that the input is either a search request or an IP address/ domain name. Nowhere does this passage teach or even suggest the feature of *receiving a search statement as a result of a user input, wherein the search statement includes a universal resource identifier and a regular expression*. Rather, *Schneider* teaches that the input is either a domain name or IP address or a search request, but not both. Thus, the passages cannot teach the feature of *receiving a search statement as a result of a user input, wherein the search statement includes a universal resource identifier and a regular expression*. Thus, *Schneider* fails to anticipate claim 1, as *Schneider* fails to teach each and every feature of claim 1, as recited in claim 1.

Additionally, *Schneider* fails to teach the feature of *parsing the retrieved universal resource identifiers for the regular expression to form search results*. The Final Office Action alleges that *Schneider* teaches this feature in column 21, lines 48-63, column 30, lines 30-42, reproduced below for the Board's convenience, and column 18, lines 30-55, reproduced above.

To generate a new URI, the parsed components (as illustrated in FIG. 2b) are concatenated with a resolvable TLD. For example a fictitious URI such as "http://united.states" is parsed, where "http" is the scheme **262**, there is no path **268** or port **274**, the hostname **282** is "united", and the HLD **280** is "states". The HLD is determined in step **310** to be a TLDA, which is then modified by recursive truncation to yield the resolvable TLD "st". The parsed components; scheme **262**, hostname **282**, and TLD can then be concatenated with the "." delimiter to generate in step **422** a new URI "http://united.st". Another example, "http://united.stores" would also yield "http://united.st". Truncation enables a correspondence between abbreviations and names or words. This method yields know-how for new and creative purchasing strategies of domain names from countries around the globe.

Though a user can adjust configuration settings (as discussed in conjunction with FIGS. 7, 8, 9, 10a, 11) in lieu of accessing a registry or translation database for URI generation through different methods of calculation/resolution, such configurations can be cumbersome adding extra steps for the user to obtain desired results. Accordingly, FIG. 14a shows how a registry is used (in lieu of configuration settings) instead to generate a valid URI (step **426**). First, a registered resolvable TLD is

retrieved in step **1410** from a matching registration record of the registered name in the registry **182** and then a valid URI is calculated and generated in step **1414** from the parsed input **210** and retrieved resolvable TLD.

The passage in column 21, lines 48-63 of *Schneider* teaches that parsed components are concatenated with a resolvable top-level domain (TLD) to form a new URI. This only occurs when the input received does not contain a valid URI, as explained in *Schneider*, column 19, line 66 through column 20, line 31, reproduced below for the Board's convenience.

When input **210** includes a domain name, resolvability can be determined by parsing a HLD from the input **210**. Referring now to the prior art of FIG. 2b, a URI **210'** including a scheme, Fully Qualified Domain Name (FQDN), port, and path is parsed. The scheme **262** is parsed in step **260** from the URI **210'** leaving the FQDN, port, and path **264**. The path **268** is parsed in step **266** from the FQDN, port, and path **264** leaving the FQDN and port **270**. The port **274** is parsed in step **272** from the FQDN and port **270** leaving the FQDN **276**. The HLD **280** is parsed in step **278** from the FQDN **276** leaving a hostname **282**. FIG. 2b illustrates one of many parsing schemes that can be applied when parsing input in step **210**.

Rather than displaying an error message in step **230** in response to input **210** having a domain name determined in step **226** to be not valid (e.g., fictitious domain name), in a preferred aspect of the present invention, further steps are performed instead as shown in FIG. 3. The HLD **280** is compared in step **310** to a list of resolvable TLDs **180** to determine HLD **280** resolvability in step **314**. If the HLD **280** is determined in step **314** to be resolvable, then a more specific browser error message **318** can be displayed stating "Domain name is not valid. Select link to learn more about proper domain name syntax". Upon display, it can further be determined in step **322** whether received input **210** is processed as a search request in step **218**. When this is the case, a search request is processed and results if any, can then be notified, accessed, and/or displayed in step **222**. When the HLD **280** is determined in step **314** to be not resolvable, then the HLD **280** is a TLDA, and the input **210** now determined to have a FDN is further processed in step **326**. After TLDA processing in step **326**, the step of URI resolvability can be determined in step **242**.

The above-cited passage of *Schneider teaches that the URI received as the input search statement, is parsed*. Then as explained in column 21, lines 48-63, these parsed components of the URI received as input for a search statement, are then concatenated to try and generate a new, valid URI.

The passage in column 30, lines 30-42 merely teaches how to use a registry in lieu of configuration settings to generate a valid URI to be used in place of the invalid URI supplied as the search input.

Thus, these two passages of *Schneider* merely teach methods for generating a valid URI to use in a search when an invalid URI is supplied as a search input.

As discussed above, the passage in column 18 of *Schneider* merely teaches the steps of requesting a network resource by using resource location and/or name resolution services.

Thus, none of the passages cited by the Final Office Action as allegedly teaching the feature of *parsing the retrieved universal resource identifiers for the regular expression to form search results*, actually teach the feature of *parsing the retrieved universal resource identifiers for the regular expression to form search results*, as none of these passage teaches anything about a universal resource identifier that is retrieved as a result of a search.

Furthermore, none of the cited passages teaches a regular expression or parsing a retrieved universal resource identifier for a regular expression. Rather, *Schneider* is directed towards an improved method for generating domain names. Thus, if a domain name entered is determined to be invalid, *Schneider* teaches a method for generating a valid domain name based on the invalid domain name. As such, *Schneider* does not teach, suggest, or even hint at retrieving URIs and then parsing the retrieved URIs to find a match to the input regular expression. Further, as *Schneider* teaches that the originally received input cannot contain both a regular expression and a URI, it follows that *Schneider* cannot teach or suggest the feature of *parsing the retrieved universal resource identifiers for the regular expression to form search results*. Thus, *Schneider* fails to anticipate claim 1, as *Schneider* fails to teach each and every feature of claim 1, as recited in claim 1.

Therefore, for at least the reasons set forth above, Appellants submit that claim 1 is in condition for allowance over the *Schneider* reference, as the *Schneider* reference fails to teach each and every feature of claim 1. Thus, the *Schneider* reference fails to anticipate claim 1.

Claims 7, 8, and 14 recite features similar to those of claim 1. Therefore, the same distinctions that distinguish claim 1 from the *Schneider* reference apply to claims 7, 8, and 14. Thus, Appellants submit that claims 7, 8, and 14 are also in condition for allowance over the *Schneider* reference.

Since claims 2-6 and 9-13 depend from claims 1 and 8, the same distinctions between *Schneider* and the claimed invention in claims 1 and 8 exist for claims 2-6 and 9-13. Therefore, Appellants submit that claims 2-6 and 9-13 are also in condition for allowance, at least by their virtue of depending from an allowable claim. Additionally, claims 5 and 12 claim other additional combinations of features not suggested by the reference.

Claims 5 and 12 recite the feature of *searching a table of contents for a match to the regular expression, wherein the table of contents contains the retrieved universal resource identifiers*. The Final Office Action alleges that *Schneider* in Figure 13, which the Final Office Action admits is a table of generated URIs, teaches this feature. (See Final Office Action mailed August 10, 2007, p. 7.) As was discussed above in regards to claim 1, *Schneider* teaches generating URIs based on the parsed components of an invalid URI that was supplied as a search input. Thus, the URIs contained in Figure 13 are not *retrieved universal resource identifiers*, as recited in claims 5 and 12. Further, nothing in Figure 13, or in the text of *Schneider* that describes Figure 13, teaches, suggests, or even hints at searching the URIs for a match to a regular expression. Thus, *Schneider* fails to anticipate claims 5 and 12, as *Schneider* fails to teach each and every feature of claims 5 and 12, as recited in claims 5 and 12.

Thus, the Appellants respectfully request that the Examiner's final rejection of claims 1-14 under 35 U.S.C. § 102 be reversed.

A.2. 35 U.S.C. § 102, Anticipation (Claim 15)

Regarding independent claim 15, the Final Office Action recites:

15. A method in a data processing system for searching for Web pages within a Web site, the method comprising:
 - receiving a search statement from a user at a client browser, wherein the search statement includes a universal resource identifier and a regular expression;
 - in response to receiving the search statement at the client browser, sending a request, by the client browser, to a server to retrieve a table of contents, wherein the table of contents comprises universal resource

identifiers associated with the universal resource identifier in the request; receiving the table of contents from the server;

 parsing the universal resource identifiers in the received table of contents for the regular expression, by the client browser, to form search results; and

 displaying the search results to the user, wherein the search results include a list of universal resource identifiers associated with the Web pages within the Web site .

Claim 15 recites features similar to those of claim 1. Additionally, claim 15 recites the feature of “in response to receiving the search statement at the client browser, sending a request, by the client browser, to a server to retrieve a table of contents, wherein the table of contents comprises universal resource identifiers associated with the universal resource identifier in the request.” The Final Office Action alleges that *Schneider* teaches this feature in column 19, line 65 through column 20, line 32, and column 30, lines 22 through 28, both of which passages are reproduced above.

However, neither of these passages of *Schneider* teaches this feature. The passage of *Schneider* in column 19, line 65 through column 20, line 32, merely teaches parsing the received input. In contradistinction, claim 15 recites retrieving a table of contents from a server, wherein the table of contents is associated with the URI in the received request. The passage of *Schneider* in column 19, line 65 through column 20, line 32, does not teach or even hint at, a server, a table of contents, retrieving a table of contents from a server, a set of URIs associated with the URI in the received input, or that a retrieved table of contents contains a set of URIs associated with the URI received as part of the initial input.

. The passage of *Schneider* in column 30, lines 22 through 28, discusses a table that shows an input domain name/URI and the **results that can be generated based on parsing the input domain name**. Again, this passage of *Schneider* fails to teach or even hint at, a server, a table of contents, retrieving a table of contents from a server, a set of URIs associated with the URI in the received input, or that a retrieved table of contents contains a set of URIs associated with the URI received as part of the initial input.

Thus, *Schneider* fails to teach the feature of “in response to receiving the search statement at the client browser, sending a request, by the client browser, to a server to retrieve a table of contents, wherein the table of contents comprises universal resource identifiers associated with the universal resource identifier in the request.”

Therefore, for at least the reasons set forth above and in Section A.1., Appellants respectfully submit that claim 15 is in condition for allowance over the *Schneider* reference. Thus, the Appellants respectfully request that the Examiner's final rejection of claims 15 under 35 U.S.C. § 102 be reversed.

/Gerald H. Glanzman/

Gerald H. Glanzman

Reg. No. 25,035

YEE & ASSOCIATES, P.C.

PO Box 802333

Dallas, TX 75380

(972) 385-8777

GG/blj

CLAIMS APPENDIX

The text of the claims involved in the appeal are:

1. A method in a data processing system for searching for Web pages within a Web site, the method comprising:

receiving a search statement as a result of a user input, wherein the search statement includes a universal resource identifier and a regular expression;

retrieving universal resource identifiers associated with the universal resource identifier in the request to form retrieved universal resource identifiers;

parsing the retrieved universal resource identifiers for the regular expression to form search results; and

returning the search results, wherein the search results include a list of universal resource identifiers associated with the Web pages within the Web site.

2. The method of claim 1, wherein the search results are returned as a Web page, wherein the universal resource identifiers are presented as a set of links, wherein selection of a link within the set of links causes a Web page identified by the link to be retrieved.

3. The method of claim 1, wherein the regular expression is separated from the universal resource identifier by a delimiter.

4. The method of claim 1, wherein the universal resource identifier is a domain name.

5. The method of claim 1, wherein the parsing step includes:
searching a table of contents for a match to the regular expression, wherein the table of contents contains the retrieved universal resource identifiers.
6. The method of claim 1, wherein retrieving, parsing, and returning steps are performed by a server hosting a Web site identified by the universal identifier, a proxy server, or a client at which the user input was entered.
7. A data processing system for searching for Web pages within a Web site, the data processing system comprising:
a bus system;
a communications unit connected to the bus system;
a memory connected to the bus system, wherein the memory includes a set of instructions;
and
a processing unit connected to the bus system, wherein the processing unit executes the set of instructions to receive a search statement as a result of a user input in which the search statement includes a universal resource identifier and a regular expression;
retrieve universal resource identifiers associated with the universal resource identifier in the request to form retrieved universal resource identifiers; parse the retrieved universal resource identifiers for the regular expression to form search results; and return the search results in which the search results include a list of universal resource identifiers associated with the Web pages within the Web site.

8. A data processing system to search for Web pages within a Web site, the data processing system comprising:

receiving means for receiving a search statement as a result of a user input, wherein the search statement includes a universal resource identifier and a regular expression;

retrieving means for retrieving universal resource identifiers associated with the universal resource identifier in the request to form retrieved universal resource identifiers;

parsing means for parsing the retrieved universal resource identifiers for the regular expression to form search results; and

returning means for returning the search results, wherein the search results include a list of universal resource identifiers associated with the Web pages within the Web site.

9. The data processing system of claim 8, wherein the search results are returned as a Web page, wherein the universal resource identifiers are presented as a set of links, wherein selection of a link within the set of links causes a Web page identified by the link to be retrieved.

10. The data processing system of claim 8, wherein the regular expression is separated from the universal resource identifier by a delimiter.

11. The data processing system of claim 8, wherein the universal resource identifier is a domain name.

12. The data processing system of claim 8, wherein the parsing means includes:
searching means for searching a table of contents for a match to the regular expression,
wherein the table of contents contains the retrieved universal resource identifiers.

13. The data processing system of claim 8, wherein retrieving, parsing, and returning means are performed by a server hosting a Web site identified by the universal identifier, a proxy server, or a client at which the user input was entered.

14. A computer program product in a computer readable medium for searching for Web pages within a Web site, the computer program product comprising:
first instructions for receiving a search statement as a result of a user input, wherein the search statement includes a universal resource identifier and a regular expression;
second instructions for retrieving universal resource identifiers associated with the universal resource identifier in the request to form retrieved universal resource identifiers;
third instructions for parsing the retrieved Web pages for the regular expression to form search results; and
fourth instructions for returning the search results, wherein the search results include a list of universal resource identifiers associated with the Web pages within the Web site.

15. A method in a data processing system for searching for Web pages within a Web site, the method comprising:
receiving a search statement from a user at a client browser, wherein the search statement includes a universal resource identifier and a regular expression;

in response to receiving the search statement at the client browser, sending a request, by the client browser, to a server to retrieve a table of contents, wherein the table of contents comprises universal resource identifiers associated with the universal resource identifier in the request; receiving the table of contents from the server; parsing the universal resource identifiers in the received table of contents for the regular expression, by the client browser, to form search results; and displaying the search results to the user, wherein the search results include a list of universal resource identifiers associated with the Web pages within the Web site.

EVIDENCE APPENDIX

There is no evidence to be presented.

RELATED PROCEEDINGS APPENDIX

There are no related proceedings.